AMENDMENTS TO THE CLAIMS

1.   (Currently amended) A process for reordering items in a database to be retrieved for display to a user, comprising the steps of:

accepting user input from a keyboard;

~~providing~~ ordering words in a linguistic database (LDB) ~~containing words ordered~~ according to a linguistic frequency of use model;

~~providing~~ ordering words in a user database (UDB) separate from the LDB ~~which~~ that stores user defined words that the user ~~specifically~~ enters into the system and ~~which~~ that includes a reorder database (RDB)~~;~~

storing in the RDB ~~that stores database~~ object numbers for words contained in the LDB;

dynamically retrieving from said LDB and from said UDB words that include the sequence of letters formed by the user input;

displaying to the user a list of ~~said~~ retrieved words, wherein the order of words determined by the RDB are displayed before the order of words determined by the LDB ~~ordered using the linguistic database ordering overridden by any existing dynamic reordering frequency count~~;

enabling the user to select a word from the displayed list; and

assigning a dynamic reordering frequency count to words selected by the user and inserting the selected words' assigned reordering frequency counts ~~and object numbers into said reorder database~~ into the RDB.

2.-3.   (Canceled)

4.   (Previously Presented) The process of Claim 1, wherein said assigning step inserts into said reorder database a non first ordered word from said list if the user selects the non first ordered word for the first time, and inserts into said reorder database the nonselected first ordered word if it does not already exist in said reorder database.

5.   (Previously Presented) The process of Claim 4, wherein if the non first ordered word is selected by the user a predetermined number of times then the non first ordered word is assigned a higher frequency count than the first ordered word.

6.   (Previously Presented) The process of Claim 4, wherein all non first ordered words entered into said reorder database are initially assigned equal reordering frequency counts.

7.   (Previously Presented) The process of Claim 1, wherein a word's reordering frequency count is increased each time the user selects the word.

8.   (Previously Presented) The process of Claim 1, wherein if the user selects from said list a word which is below a second ordered position then said assigning step assigns a value to the word's reordering frequency count that places the word in the second ordered position in said list.

9.   (Previously Presented) The process of Claim 1, further comprising the step of:
        periodically aging the reordering frequency counts in said reorder database by reducing the reordering frequency counts by a predetermined factor.

10.   (Previously Presented) The process of Claim 1, further comprising the step of:
        periodically checking free space in said reorder database and, if the free space is less than a predetermined amount, then removing from said reorder database words that have reordering frequency counts below a predetermined threshold.

11.   (Previously Presented) The process of Claim 10, wherein said removing step removes user defined words having reordering frequency counts below the predetermined threshold after removing other words having reordering frequency counts below the predetermined threshold.

12.   (Previously Presented) The process of Claim 1, further comprising the step of:

resolving reordering frequency collisions when two words in said list have equal reordering frequency counts by ordering the most recently selected of the two words first.

13.   (Previously Presented) The process of Claim 1, further comprising the step of:

resolving reordering frequency collisions when two words in said list have equal reordering frequency counts by ordering the word having a higher initial ordering in said linguistic database first.

14.   (Previously Presented) The process of Claim 1, further comprising the step of:

resolving reordering frequency collisions in said list when a user defined word and a word from said linguistic database have equal reordering frequency counts by ordering the user defined word first.

15.   (Original) The process of Claim 1, wherein words selected by the user that do not have a possibility of a collision with other words are not assigned a reordering frequency count.

16.   (Currently amended) An apparatus for reordering items in a database to be retrieved for display to a user, comprising:

a module for accepting user input from a keyboard;

a linguistic database (LDB) containing words ordered according to a linguistic frequency of use model;

~~providing~~ a user database (UDB) separate from the LDB ~~which~~ that stores user defined words that the user ~~specifically~~ enters into the ~~system~~ module and ~~which~~ that includes a reorder database (RDB) that stores database object numbers for words contained in the LDB;

a module for dynamically retrieving from ~~said~~ the LDB and from ~~said~~ the UDB words that include the sequence of letters formed by the user input;

a module for displaying to the user a list of ~~said~~ retrieved words ~~ordered using the linguistic database ordering overridden by any existing dynamic reordering frequency~~

5

~~count~~ , wherein the order of words determined by the RDB are displayed before the order of words determined by the LDB;

enabling the user to select a word from the displayed list; and

a module for assigning a dynamic reordering frequency count to words selected by the user and inserting the selected words' assigned reordering frequency counts ~~and object numbers into said reorder database~~ into said RDB.

17.-18. (Canceled)

19.   (Previously Presented) The apparatus of Claim 16, wherein said assigning module inserts into said reorder database a non first ordered word from said list if the user selects the non first ordered word for the first time, and inserts into said reorder database the nonselected first ordered word if it does not already exist in said reorder database.

20.   (Previously Presented) The apparatus of Claim 19, wherein if the non first ordered word is selected by the user a predetermined number of times then the non first ordered word is assigned a higher frequency count than the first ordered word.

21.   (Previously Presented) The apparatus of Claim 19, wherein all non first ordered words entered into said reorder database are initially assigned equal reordering frequency counts.

22.   (Previously Presented) The apparatus of Claim 16, wherein a word's reordering frequency count is increased each time the user selects the word.

23.   (Previously Presented) The apparatus of Claim 16, wherein if the user selects from said list a word which is below a second ordered position then said assigning module assigns a value to the word's reordering frequency count that places the word in the second ordered position in said list.

24.   (Previously Presented) The apparatus of Claim 16, further comprising:

a module for periodically aging the reordering frequency counts in said reorder database by reducing the reordering frequency counts by a predetermined factor.

25.   (Previously Presented) The apparatus of Claim 16, further comprising:

a module for periodically checking free space in said reorder database and, if the free space is less than a predetermined amount, then removing from said reorder database words that have reordering frequency counts below a predetermined threshold.

26.   (Previously Presented) The apparatus of Claim 25, wherein said removing module removes user defined words having reordering frequency counts below the predetermined threshold after removing other words having reordering frequency counts below the predetermined threshold.

27.   (Previously Presented) The apparatus of Claim 16, further comprising:

a module for resolving reordering frequency collisions when two words in said list have equal reordering frequency counts by ordering the most recently selected of the two words first.

28.   (Previously Presented) The apparatus of Claim 16, further comprising:

a module for resolving reordering frequency collisions when two words in said list have equal reordering frequency counts by ordering the word having a higher initial ordering in said linguistic database first.

29.   (Previously Presented) The apparatus of Claim 16, further comprising:

a module for resolving reordering frequency collisions in said list when a user defined word and a word from said linguistic database have equal reordering frequency counts by ordering the user defined word first.

30.   (Original) The apparatus of Claim 16, wherein words selected by the user that do not have a possibility of a collision with other words are not assigned a reordering frequency count.

31.   (Previously Presented) The process of Claim 1 wherein when a word from the LDB is selected for the first time, said step of assigning uses the word's frequency of use order in the LDB as an initial dynamic reordering frequency count.

32.-33. (Canceled)

34.   (Previously Presented) The process of Claim 31, wherein all words specifically entered by the user are initially assigned equal reordering frequency counts by said assigning step.

35.-44. (Canceled)

45.   (Previously Presented) The apparatus of Claim 16 wherein when a word from the LDB is selected for the first time, said module for assigning uses the word's frequency of use order in the LDB as an initial dynamic reordering frequency count.

46.-47. (Canceled)

48.   (Previously Presented) The apparatus of Claim 45, wherein all words specifically entered by the user are initially assigned equal reordering frequency counts by said assigning module.

49.-60. (Canceled)